Authentication using Audio Key Phrase Integrated with Random Number Generated Keypad

Herny Ramadhani Mohd Husny Hamid[1], Norhaiza Ya Abdullah[2], Wan Hazimah Wan Ismail[3]
Universiti Kuala Lumpur Malaysian Institute of Information Technology
1016, Jalan Sultan Ismail, 50250 Kuala Lumpur
herny@unikl.edu.my[1], norhaizaya@unikl.edu.my[2], wanhazimah@unikl.edu.my[3]

ABSTRACT

Most organizations have a tendency to focus on preventing threats from external attacks and almost overlooked on internal attacks. Biometrics, access control cards, keypad controls and door-locks are common types of physical security countermeasures. Failure of such design may endanger and post serious damage to major economics infrastructure, personal privacy and facilitate crime. Strong physical security applies the principle of Defense in Depth where multiple layers of protection are used in strengthening the security. This paper proposed an alternative security system design which uses audio as first layer of authentication to identify key-phrases generated randomly by the system with the integration of Random Number Generated (RNG) keypad as the second layer of authentication. Furthermore this system has the ability to resists man in the middle attack, and insider attacks because the system is connected in isolated and separated network, uses key-phrases identified from random audio and use software based keypad, with randomize number position. It is hoped that by having this system, common physical security countermeasure weaknesses can be minimized.

KEYWORDS

Security, Authentication, Audio, Random Number Generated, One Time Password.

1 INTRODUCTION

Securing protected area has becoming more challenging tasks these days. Security engineers need to implement many security measurements to provide better security approach. The attempts on breaking or by passing the security measurements created are increasing as well as the improvement of the security measurement.

Without strong physical security an organization has to spend thousands of dollars on anti-virus, firewalls, and intrusion prevention systems only to have confidential data stolen by a careless error [1].

The following are main methods of authentications [2]:

a. Knowledge based authentication techniques are the most widely used authentication techniques but it is vulnerable to many types of attack such as dictionary attacks, brute-force attacks, spyware, social engineering, shoulder surfing [3].
b. Token based authentication techniques, such as key cards, bank cards and smart cards are widely used. Many token-based authentication systems also use knowledge-based techniques to enhance security.
c. Biometrics based authentication techniques, such as fingerprints, iris scans, or facial recognition has been developed due to unique properties of biometrics. The major drawback of this approach is that such systems can be expensive, and the identification process can be slow and often unreliable. People tend to forget their passwords [4], [5] due to human memory's fallibility and reminders or replacements are needed. Cost of replacement is anything but negligible and has to be funded. Some users tend to use unsafe practices like writing them down, saving it in email drafts,

personal computers, reusing the same password across multiple sites, or frequently reinitializing passwords upon failure to authenticate [6], [7], [8], [9].

In addition, with advancement of electrical appliance and the increased of mechanical locks break in electronics locks issues have become well accepted in 1970s. Hotels in the year of 1970 have been steadily using card locks [6]. This is due to the diversity of electronic locks products which have all sorts of mechanism from contactless smart cards through PIN pads to biometrics.

Not long after that biometrics locks came into the market which was able to provide better security policy and access control list management, but most of the time this sophisticated biometrics devices are costly [6]. Hand biometric is easy to steal, which victims may leave their fingerprints everywhere they touched, retinal scan everywhere human look, even fingerprints on ID card maybe stolen by a hacker and used for other purposes. The worst of all, the victims cannot change or update their biometrics because passwords can be changed and updated but biometrics cannot be changed as it is permanent. If a hacker has copied the victim thumbprint, the victim is out of luck.

In this paper, a new security system design which applies the principle of Defense in Depth has been proposed. This system uses audio to identify key-phrases as first layer of authentication, which is generated randomly by the system with the integration of Random Number Generated (RNG) keypad as a second layer of authentication. User will listen to random audio files and they will need to identify the correct key-phrases by pressing the button at the headset provided. On each button pressed, system will sync with the server for verification process. Once all key-phrases are verified, user needs to enter their secret pin through RNG keypad. Failure of identifying single key-phrase will result to restart of the process. On third failure attempt, the system will be locked down and new key-phrases will be sent to the user via SMS and email.

This paper is structured as follows. In Section 2, related works on common methods used and specifically on the audio key phrase authentication, random number generated keypad and one time password is described. Section 3 reviews the development process of the prototype system. The testing results based on the two methods which were user acceptance testing, and black box testing is discussed in section 4. The following section deliberated on the security analysis based on several attack of the proposed system. Finally, the conclusion, and the future improvement were discussed.

2 RELATED WORKS

2.1 Common Authentication Methods

Many methods can be used for authentication process but in order to choose the suitable methods for the authentication process, there are a few factors that the researchers preferred. Table 1 shows the comparison of common authentication methods.

Table 1. Comparison of authentication methods

Factor	Cost Effective	Secure Level	Drawback
Keycard	Not	Medium	Must carry the token
Password	Yes, but depends on its implementation	Low	Must memorize password
Biometric	Not	High	Must exactly same
Digital Audio	Yes	High	Must exactly same
Prototype System	Yes	High	None

2.2 Audio Key Phrase Authentication

In utilizing audio as authentication mechanism in android platform, Text-to-Speech audio generator has been used. Text-to-Speech audio generator is a function available in android platform since version 1.6 which is known as Text-to-Speech (T.T.S) or also known as speech synthesis where T.T.S enables android device to speak text of different language. To use T.T.S, the T.T.S engine needs to know which language is used. T.T.S is able to pronounce "Paris" differently in French and English as long as the android device is installed with the T.T.S specific support language. The process of Text-to-Speech is shown in Figure 1.

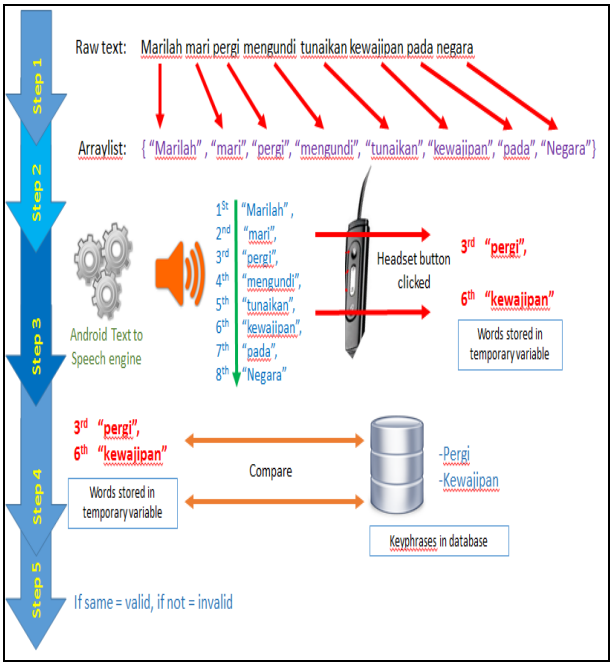

Figure 1. Text-to-Speech processes

Each word in the raw text is in an array list. Assume the key phrase is "*pergi*" and "*kewajipan*", user is required to click headset button once he/she heard the word. These two words then will be stored in temporary variable to be compared with the key phrase stored in the database. It is valid if the word is the same with the key phrase in the database. If not, user is required to do the same process once again.

From Table 2, it shows that by using Audio time frame approach, CPU's resources consumption is low while complexity in developing the system is medium.

Table 2. Comparison of methods for Audio authentication process

Factors	**Audio time frame matching**	**Text speech audio generator**
CPU resources consumption	Low	Medium
System complexity	Medium	Medium
Possible to be develop in android software development	Possible	Possible

2.3 Random Number Generated (RNG) Keypad

RNG keypad is a computational or physical device designed to generate a sequence of numbers or symbols that lack any pattern, i.e. appear random. Random number generators have applications in cryptography, gambling, statistical sampling, computer simulation, completely randomized design, and other areas where producing an unpredictable result is desirable.

Generally, where unpredictability is paramount such as in security applications, hardware generators are generally preferred (where feasible) over pseudo-random algorithms. Table 3 shows two principal methods used to generate random numbers which is Pseudo-random and True-random.

Table 3. Comparison between Pseudo-Random and True-Random

Principle vs. Methods	**Pseudo-random**	**True-random**
Approach	Algorithm of mathematical formula, later translated into relatively bits of programming code	Extract randomness from physical phenomena and introduce it into a computer

Efficiency	Fast responses in generating numbers	Slow responses in generating numbers
Determinism	Sequence of numbers can be reproduced	Sequence of numbers cannot be reproduced
Periodicity	Sequence of numbers is repeated	Sequence of numbers will or will not repeated

2.4 ONE TIME PASSWORD (OTP)

OTP is a password authentication scheme in which a new password is generated for each authentication session. Once the password is used, it is no longer valid and any attempt to reuse the same password for future authentication sessions will fail [10]. OTPs avoid a number of shortcomings that are associated with traditional (static) passwords.

The most important shortcoming that is addressed by OTPs is that, in contrast to static passwords, they are not vulnerable to replay attacks. This means that a potential intruder who manages to record an OTP which have already been used to log into a service or to conduct a transaction will not be able to abuse it, since it will be no longer valid. On the downside, OTPs are difficult for human beings to memorize. OTP generation algorithms typically make use of pseudo-randomness or true-randomness. This is necessary because otherwise it would be easy to predict future OTPs by observing previous ones.

2.4.1 Time-synchronize One Time Password

A time-synchronized OTP is usually related to a piece of hardware called a security token (e.g., each user is given a personal token that generates a one-time password). Inside the token is an accurate clock that has been synchronized with the clock on the proprietary authentication server.

On these OTP systems, time is an important part of the password algorithm, since the generation of new passwords is based on the current time rather than, or in addition to the previous password or a secret key. This token may be a proprietary device, or a mobile phone or similar mobile device which runs software that is proprietary, freeware, or open-source. An example of time-synchronized OTP standard is TOTP. All methods of delivering the OTP below may use time-synchronization instead of algorithms.

3 PROTOTYPE SYSTEMS

3.1 Audio Key Phrase Authentication

User needs to interact in order to run and use the system. In order to implement that, the two devices needed are a tablet and a headphone as shown in Figure 2.

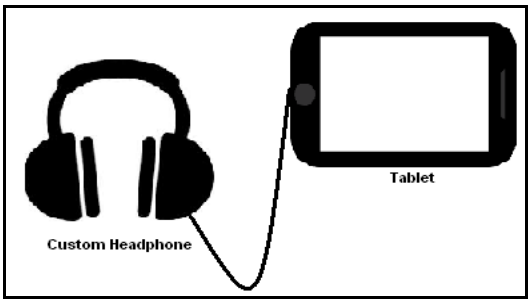

Figure 2. Connection between Headphone and Tablet

The researchers also take into account on how the user should respond to the correct key phrases when he /she heard the pre saved sounds from the system. The researchers decided to attach a small button that attached to the customized headphone instead of the user just interact directly to the tablet. This is to ensure the effort of guessing and predicting the key phrases is becoming harder where the point of attack and vulnerable point for guess and predicting is hard to be seen in casual manner.

The system then needs to be synchronized with the external database. As shown in Figure 3, the system used an external database where the database is located inside the server which stores password key phrases, and there is an authentication engine inside the server which used to authenticate user. The entire audio key phrase selected by the user will be sent to the server for authentication process. The server then replied to the system whether the user has successfully input the correct value or not.

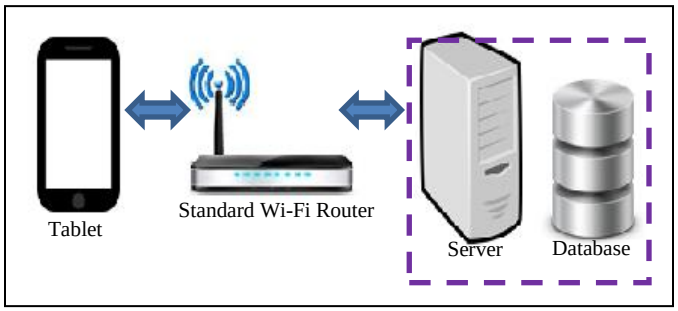

Figure 3. Audio Key Phrase Identification Block Diagram

3.2 Random Number Generated (RNG) Keypad System

RNG keypad based on OTP concept system consists of three subsystems as illustrated in Figure 4.

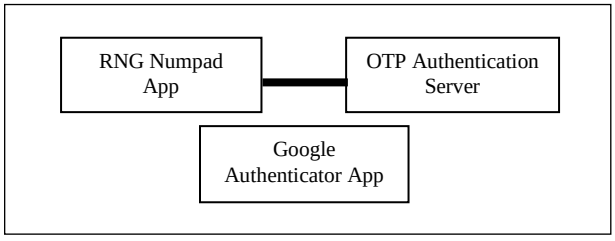

Figure 4. RNG Keypad Sub-System Diagram

3.2.1 Random Number Generator (RNG) Keypad Application Requirements

For the RNG Keypad, the aim is to build an application that can be compiled and run with the Android operating system. Android is an open-source software stack for a wide of mobile devices and a corresponding open-source project 42 led by Google.

Based from the Android developer site, it was identified that the Android application runs on Java and XML languages. Software Development Kit or SDK which includes a debugger, libraries, mobile emulator based on QEMU, documentation, sample code and tutorials are ready to be downloaded from the Android developer's site [9]. Hence, the RNG Keypad application is written using Java as it core language. Currently supported development platforms include computer which running Linux, Mac OS X 10.5.8 or later and Windows XP or later. Such platform is needed in order to develop the above application.

The term Integrated Development Environment or IDE is referring to the platform which provides comprehensive facilities to computer programmers for software development. The officially supported IDE for developing an Android app is Eclipse with Android Development Tools (ADT) plugin, the Android Studio by the Google Developer, Intellij IDEA IDE and Netbeans.

In previous works, several Java applications which used the Eclipse software have been developed, thus, Eclipse is the IDE of choices. Google Developer site offers the Eclipse software bundle with ADT plugins which can be found at their download section. This development used Google Nexus 7 tablet (Android 4.4 KitKat) as it emulator device test bed.

3.2.2 One Time Password (OTP) Authentication and User Management Server

OTP is a password that is valid for only one login session or transaction. OTPs avoid a number of shortcomings that are associated with traditional (static) passwords. Mobile OTP

which uses time-synchronization between the authentication server and client fulfils the requirement to develop this system. The OTP and User Management Server are developed using HTLM5, CSS3 along with PHP and its additional dependencies and libraries on Ubuntu 12.04 LTE.

3.2.3 Google Authenticator Android Applications

According to the Google Authenticator (GA) documentation, GA generates an 80-bit secret key for each user. This is provided as a 16 characters base32 string or as a QR code. The client creates an HMAC-SHA1 using this secret key. The message that is HMAC-ed can be the number of 30 seconds periods having elapsed since the UNIX epoch or the counter that is incremented with each new code [9]. A portion of the HMAC is extracted and converted to a 6 digits code.

Pseudo code for Time-based OTP is:

```
function
GoogleAuthenticatorCode(string
secrect)
      key := base32decode(secret)
      message :=  floor(current Unix
time / 30)
      hash := HMAC-SHA1(key, message)
      offset := last nibble of hash
      truncatedHash :=
hash[offset..offset+3]
      //4 bytes starting at offset
      Set the first bit of
truncatedHash to zero
//remove the most significant bit
      code := truncatedHash mod
1000000 pad code with 0 until length
of code is 6
      return code
```

While Pseudo code for Event/Counter OTP is:

```
function
GoogleAuthenticatorCode(string
secrect)
      key := base32decode(secret)
      message := counter encoded on 8
bytes
      hash := HMAC-SHA1(key, message)
      offset := last nibble of hash
      truncatedHash :=
hash[offset..offset+3]
//4 bytes starting at offset
      Set the first bit of
truncatedHash to zero
//remove the most significant bit
      code := truncatedHash mod
1000000 pad code with 0 until length
of code is 6
      return code
```

3.2.4 Random Number Generated (RNG) Keypad

RNG Keypad consists of two main functions which are scrambling the keypad and storing the desired pin values. Code segment for Keypad Scrambler is:

```
function scramble()
(int)random(pin value)
setPosition(rectangle[x][pin value],
rectangle[y][pin value]
setImage()
setId(pin value)
```

While code segment for pin value insertion is:

```
Function insert(int pin)
Array[101]
// array to store the pin 0-9
I,=array.length
Array[i] = {pin}
Return array[i]
```

The idea is to first scramble the position of the buttons (referring to the keypads) position along with its values, in other word, each button is assigned to its own number in the First Round. Next, when user presses a button the positions of each button should be scrambled again in the *nth* Round and the value of the pressed button should be inserted into array. Later, when the *Enter* button is pressed, this array will be iterated and should send all the pins for authentication. This process is illustrated in Figure 5.

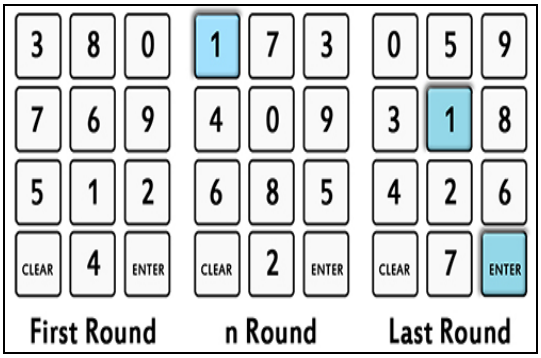

Figure 5. Scrambler and Insertion Illustration

3.2.5 Grafical User Interface (GUI) Design

The RNG Keypad GUI design is illustrated in Figure 6.

Figure 6. RNG Keypad Proposed Design

3.3 Complete System

As mentioned before, this system uses audio file which is pre-recorded speech for user to identify key-phrases generated randomly by the system during the user registration phase as first layer of authentication. The Random Number Generated (RNG) keypad will act as the second form of authentication.

User will listen to random audio files and they will need to identify the three key-phrases by pressing the customize button at the headset provided. On each button pressed, system will sync with the server for the verification process. Once all key-phrases have been identified, RNG keypad will be popup through the screen panel. User will need to enter his / her secret pin through this keypad.

Failure of identifying single key-phrase will result to restart of the process. On third failure attempt, the system will be locked and new key-phrases will be sent to the user via email and SMS.

Figure 7 shows the block diagram of this project.

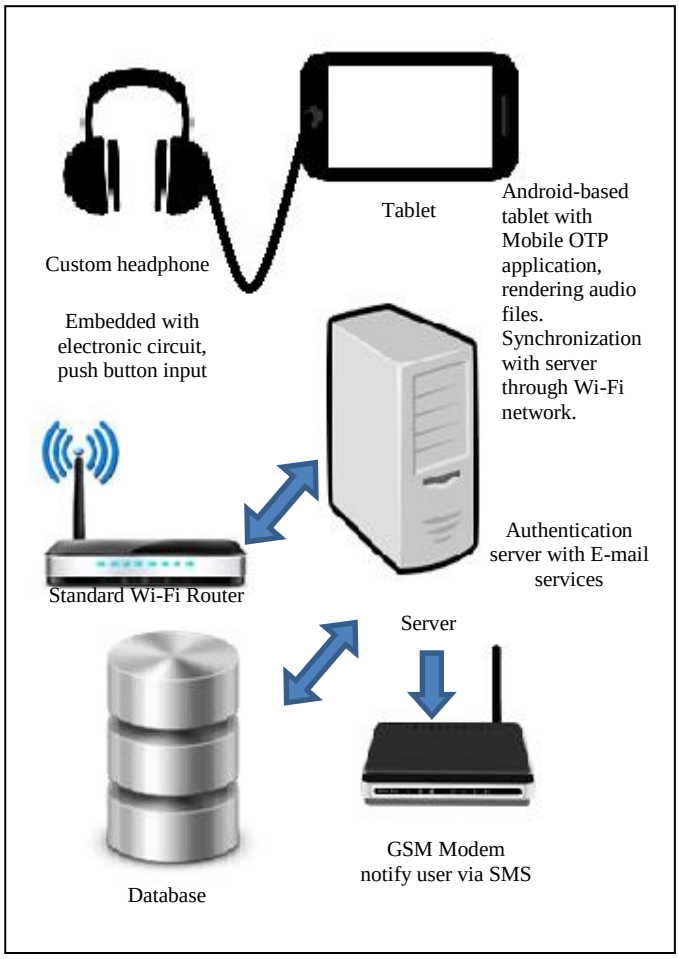

Figure 7. RNG Keypad Block Diagram

4 RESULT

The testing phase includes User Acceptance Testing (UAT) and Black Box Testing. Both tests were performed to ensure the functionality and operations of RNG keypad based on OTP concept system that met the system requirement.

4.1 User Acceptence Testing (UAT)

50 respondents were asked on the system GUI's design. The questionaire was devided into four categories; system design, understandable of system workflow, the originality and the security level of the system.

The summary of the result is illustrated in Figure 8. Overall respondents felt that it is a good system.

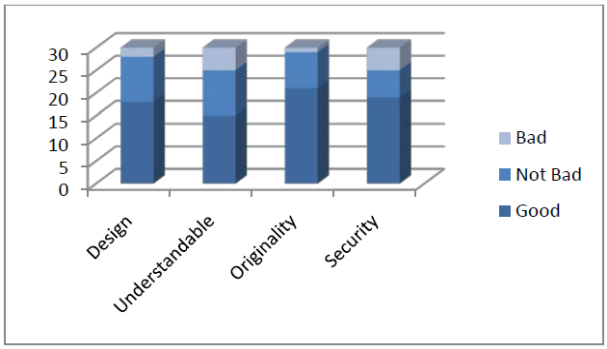

Figure 8. User feedback on UAT

4.2 Black Box Testing Result

Table 4 shows that all test cases for each subsystem and complete system is successful.

Table 4. Result of Black Box Testing

No.	Test Case and Description	Sub-system	Result
1.	Headset Connectivity: Headset button is detected and has interact with application	Audio Key Phrase identifi-cation	Pass
2.	Headset speaker: able to play audio		Pass
3.	GSM modem: able to send SMS to new registered user		Pass
	GSM modem: able to send SMS when system reset audio key phrases		Pass
4.	Server connectivity: Sending user selected key phrase to the server		Pass
5.	Integration: Able to proceed to next authentication after user select all key phrases correctly.	Hybrid Authenti cation System (HAS)	Pass
6.	User Management Server: Valided user registration	RNG Keypad based on OTP Concept	Pass
7.	Keypad Scrambler: Value and position is scrambled		Pass
8.	OTP Authentication Server: OTP value is authenticated, OTP value length is accurate, OTP value in server change every 30 seconds, have connectivity between OTP and server		Pass
9.	Verify username is valid; username and OTP pins can be authenticated	Hybrid Authenti cation System (HAS)	Pass
10.	Client/Server connectivity: valid username and OTP pins can be authenticated with and without network connectivity		Pass

5 SECURITY ANALYSIS

According to the testing results and research that have been done, this system has the ability to minimize the main attacks such as man-in-the-middle attack, and insider attack.

5.1 Man in the Middle (MITM) Attack Resistance

One of the most prevalent network attacks used against individuals and large organizations alike are man-in-the-middle (MITM) attacks. Considered an active eavesdropping attack, MITM works by establishing connections to victim machines and relaying messages between them [11]. This system is connected to an isolated network. It is separated from the main organization network and it does not

require any connection to the Internet, hence, the server will not be able to be accessed outside from its network.

5.2 Insider Attack Resistance

Another common attack of information system is insider attack. An insider can be thought of as an individual who is an employee (past or present who interacts with the system, has access to network, system or data and may disrupt reliability, integrity and approachability of the system intentionally or accidentally[12]. There are several methods used by insider to breach authentication and most of the method are using one layer of authentication Such as RFID Access Card system (125 kHz / MIFARE 13.56Mhz cards) which uses a card to authenticate, biometrics which uses thumbs, iris and etc., door locks which requires keys.

Nevertheless for our system requires two layers of authentications. First, attacker needs to know exactly set of key-phrases and second, during registering for OTP, users are required to choose their own username and password which later will be used to generate secret keys for the OTP token session. Both username and password supplied by the user acts as a SALT. The SALT will be blend with 16 digits secret keys (generated by Google Authenticator algorithm) and epoch time. Only then session token will be generated. The attacker needs to know username, password, secret keys and epoch time to replicate the OTP token.

The other attacks which fall under the insider attack were shoulder surfing, tail-gaiting, social engineering, and brute force attack. All these attacks were related with obtaining the secret information such as password. Information and computer security is supported largely by passwords which are the principle part of the authentication process [13].

Shoulder surfing is the process where the attacker can capture a password by direct observation or by recording the individual's authentication session while inserting the passwords. This attack is usually implemented in crowded places, and difficult to discern [14].

Tail-gating is another way of attacks which is often encountered in the physical world. It involves a process in which a person, whether an employee or not, passes through a secure door without the knowledge of the person who has legitimate access to the particular secure door [15].

Besides that, the attacker can also obtain the password by psychological manipulation of people into performing actions or revealing confidential information. This was refers to social engineering. For short password guessing, brute-force is the very fast method that can be used by the attacker. The common ways of entering password were faced this kind of security issues.

By comparing the common ways of entering password, this system used a set of key-phrases which user needs to identify it from an audio played through the headphone. In this case, attacker is almost impossible to know what are the audio being placed and what are the set of key-phrases are being used at that time. Audio will be selected and played randomly. In addition, the user will enter password through randomize keypad and OTP. In this case, there is no point for the attacker to peek while user entering the OTP, or guessing the OTP by memorizing fingers sequences because the password changes every 30 seconds.

This system is using a software-based keypad, with randomize number position. User will enter a random OTP number, in addition of the number position of the keypad will be scrambled once an enter button is pressed. This process will prevent the attacker from launching brute force attack.

6 CONCLUSION

This hybrid authentication system is an alternative security system design which uses two layers of authentication which is audio files to identify key-phrases and secret pin keyed in Random Number Generated (RNG) keypad. This system is in an isolated and separated network from the main network. The possible way to do remote attack is only if attacker can gain access to this isolate network. By applying MAC filtering on the filter, firewall settings on the server, other devices will not be able to connect to it. At the end of the development, the development team has succeeded in achieving all the requirements and objectives. All problems were addressed and project limitations were identified and recorded.

7 FUTURE IMPROVEMENT

The RNG Keypad is proposed to implement a vibration on each keypad pressed in order to give a soft notification to the user and the application should be developed for other mobile operating system such as IOS, Windows Mobile and Blackberry OS. As for the Audio Authentication System, a better method is needed to make the system able to understand the word contained inside any audio. The password key phrases should be encrypted because currently the password key phrases is sent to user mobile phone in plaintext. Improvement on time to authenticate (server response time) are also needed.

8 REFERENCES

[1] G.Peter and N.Stephen. (2014). *Physical Security.* Security Laboratory Article. SANS Institute

[2] K. K. Brajesh. (2012). An Approach For User Authentication One Time Password (Numeric And Graphical) Scheme, *Journal of Global Research in Computer Science, 3,* 54-57. Available: http://jgrcs.info/index.php/jgrcs

[3] A. Narayanan and V. Shmatikov. (2005). Fast dictionary attacks on passwords using time-space tradeoff. *CCS '05: Proceedings of the 12th ACM Conference on Computer and Communications Security*. ACM. (pp. 364–372)

[4] BBC News. UN warns on password 'explosion'. Available: http://news.bbc.co.uk/2/hi/technology/6199372.stm.

[5] S. Gaw and E. Felten, Password management strategies for online accounts. *Proceedings of the Symposium on Usable Privacy and Security, (SOUPS 2006),* 44-55.

[6] B. Ives, K. R. Walsh, and H. Schneider, The domino effect of password reuse. In Communications of the ACM, (CACM Apr 2004), 75-78.

[7] R. Morris and K. Thompson, Password security: A case history. Communications of the ACM (CACM Nov 1979), 594-497.

[8] M.O Rayes. (2011). *One Time Password.* Encyclopedia of Cryptography and Security (pp. 885-887)

[9] J. Steele. (2011). The Android Developer's Cookbook. Building Applications with Android SDK, pp. 2, Addison Wesley

[10] R. Anderson. (2008). Security Engineering, pp. 377, Wiley Publishing

[11] Chris Sanders. (2010). Understanding the Main in the Middle Attacks. Available: http://www.windowsecurity.com/articles-tutorials/authentication_and_encryption/Understanding-Man-in-the-Middle-Attacks-ARP-Part1.html

[12] Reza Asgari and Reza Ebrahimi (2013) A Framework To Defense Against Insider Attacks On Information Sources. *International Journal of Managing Public Sector Information and Communication Technologies (IJMPICT) Vol. 4, No. 2, June 2013.* Available: http://airccse.org/journal/mpict/papers/4213ijmpict01.pdf

[13] A. H. Lashkari, S. Farmand, O. Zakaria, and R. Salleh, (2009). Shoulder Surfing Attack in Graphical Password Authentication, *International Journal of Computer Science and Information Security, Vol. 6, No. 2.* Available: http:/ http://arxiv.org/ftp/arxiv/papers/0912/0912.0951.pdf

[14] Y. Kita, F. Sugai, M. Park, and N. Okazaki, (2013). Proposal and its Evaluation of a Shoulder-Surfing Attack Resistance Authentication Method: Secret Tap with Double Shift, *International Journal of Cyber-Security and Digital Forensics (IJCSDF) 2(1): 48-5548.* Available: http://sdiwc.net/digital-library/web-admin/upload-pdf/0000600.pdf

[15] P. Marikkannu. (2011). A Secure Mobile Agent System against Tailgating Attacks, *Journal of Computer Science 7 (4): 488-492.* Available: http:// thescipub.com/PDF/jcssp.2011.488.492.pdf